Clustering

Cluster Analysis

- **Partitioning** a set of data objects into subsets or **clusters**
    - objects in a cluster are similar, yet dissimilar to objects in other clusters

انا معرفش ايه cluster اللي هاتطلع معايا فبحاول اقسم الداتا لمجموعه من cluster بحيث الداتا في cluster الواحد بتكون شبه بعض وcluster ذات نفسها بتكون مختلفه.

لو انا في شركه وعندي عملاء وبيشتروا منتجات فانا عايز اعرف مين اليوزر اللي شبه بعض عشان احطلهم اعلاناتاو منتجات معينه جنب بعض عشان اسهل عليهم.

- **Goal**: discovery of **previously unknown groups within the data**

Cluster بيستكشف حاجات نسبتها قليله في الداتا

(لو عندي مرضي بيعانوا من اعراض غير طبيعيه)

- Clusters are ***implicit classes***
- **Applications** → business intelligence, image pattern recognition, web search, biology, security

ممكن استخدم cluster في السيكيورتي لو جالي اكتر من ريكوست علي السيرفر فهاقسم الريكوستات دي الي مين اللي عايز خدمات ومين اللي عايز يعمل هاك علي السيرفر

- Clustering can be used for *preprocessing* and *outlier detection*

ممكن استخدم cluster في preprocessing وتحديد outlier

Requirements for Cluster Analysis

- **Scalability** → currently handles small datasets, uses sampling

يعني complexity بيتزيد linear مع زياده الداتا مش موجود عن cluster algorithm كتير

- **Handling different attribute types** → mostly numerical

اتعامل مع انواع attribute المختلفه (اغلب الانواع numerical)

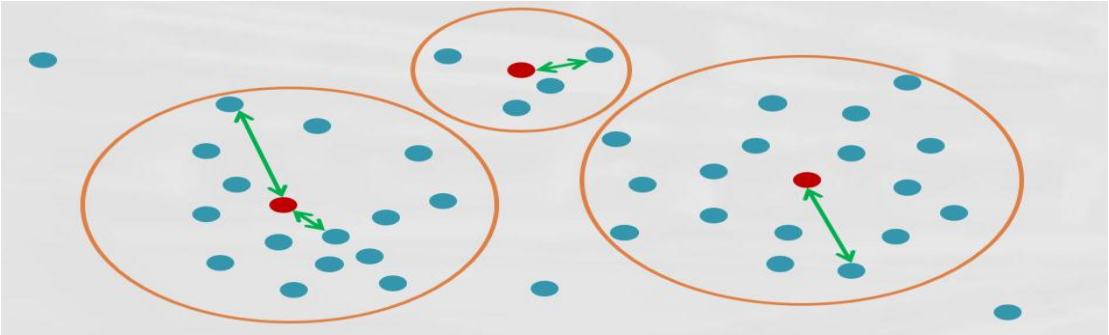
- **Discovering clusters with arbitrary shape** → currently mostly spherical

لازم اقدر اكتشف cluster باشكال مختلفه

- **Domain knowledge & input parameters** → # clusters & clustering results

لازم يبقي عندي Domain knowledge مثلا الجوريزم k-means انا اللي بدخله k فاكيد هاتاثر علي الداتا. معظم الالجوريزمات بتعتمد علي parameter فلازم اعمل test عليهم

- **Handling noisy data** → currently sensitive to noise



اي noise ممكن تاثر علي cluster shape

لو بعمل cluster لليوزر في real time مش هاقدر اتعامل معاها كويس

- **Incremental clustering & insensitivity to input order** →new data requires re-computing clusters from scratch – sensitive to order

ترتيب الداتا لو اختلف ترتيب record هايختلف الناتج معايا

- **Handling high-dimensionality data** → mostly low Dimensionality

Cluster يقدر ي handle الداتا الي 3 dimension

- **Constraint-based clustering** → little support for domain constraints

معرفش احط exception لما بحط constrain لل cluster

- **Interpretability & usability** → are results comprehensible & usable?

ممكن يطلع cluster بس مش مفهومه ومقدرش استخدمها

Comparing Cluster Analysis Methods

- **The partitioning criteria** – *flat* or *hierarchical*?

هاقسم الداتا ل (flat) يعني one layer ولا الي (hierarchical) يعني اكتر من layer

- **Separation of clusters** – *mutually exclusive* or *overlapping*?

تقسيم ال cluster
mutually exclusive: يعني اقسم ال cluster بحيث ان لو طلعت cluster معين مايطلعش معايا تاني في التقسيم

Overlapping : لو ممكن يكرر

- **Similarity measure** – *distance* or *connectivity/density*?

هايحسب similarity

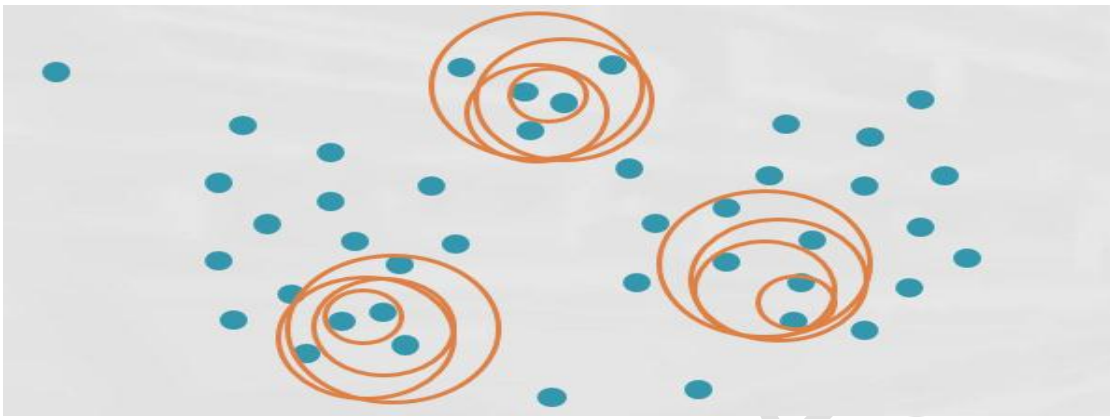
Overview of Cluster Analysis Methods

1. Partitioning

الاول هايفترض عدد cluster اللي عايزهم ويفرض مثلا ٣ نقط كل منهم في cluster ويحسب بناء عليهم كل cluster والنقط اللي فيه زي k-menas كده

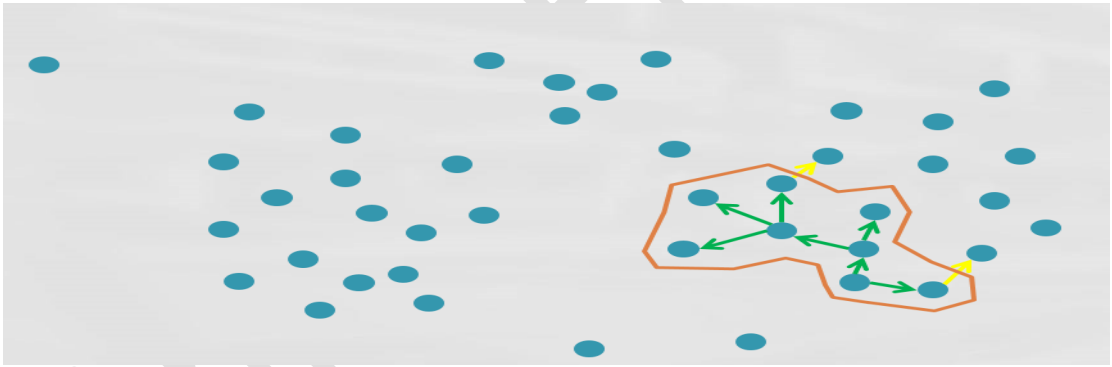
ال similarity في cluster الواحد قريبين من بعض

2. Hierarchical



هافترض ان كل item هو cluster لوحده وهابدا ادمج كل اتنين مع بعض بحيث ان الاتنين يكونوا قريبين في similarity وهكذا لحد ماخلص كل النقط

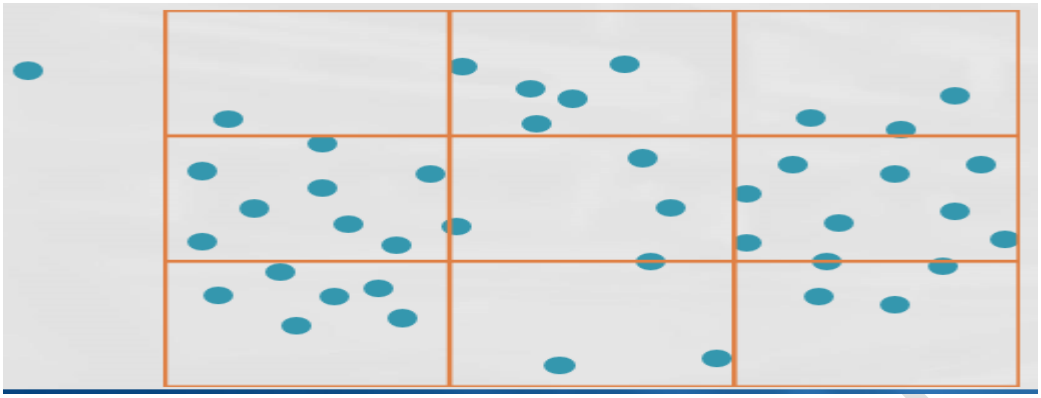
مشكله Hierarchical لو طلعت مستوي معرفش انزل منه



3. Density-based

وده بيستخدم عشان يجيب الاشكال الغير منتظمه (non linear) ومن امثلته DBSCAN
الاول بيبدا باي نقطه وبعدين ببدا احسب المسافه بينها وبين كل اللي حواليها علي حسب المسافه اللي محددها واقل عدد نقط موجود في الكلستر الواحد بتبقي متحدده قبل مااشتغل ويحسب المسافه بين كل نقطه واللي حواليها لو اللي اشوف هل كل النقط اللي حواليها واقل من المسافه اللي متحدده هل هما نفس العدد يعني لو انا محدد المسافه 2 واقل عدد في الكلستر الواحد 4 يبقي اشوف المسافه بين النقطه واللي حواليها لو لقيتهم اقل من المسافه اللي محددها وكمان يكون عددهم اكبر من او يساوي اقلل عدد انا محدده

Grid-based



ب map ال grid علي الداتا اللي عندي وبعمل fixed cluster وبخلي كل cell لوحدها

Overview of Cluster Analysis Methods

Method	Characteristics
Partitioning methods	— Find *mutually exclusive* clusters of *spherical shape* — *Distance-based* — May *use mean or medoid* to represent cluster center — Effective for *small- to medium-size data sets*
Hierarchical methods	— Clustering is *hierarchy* involving multiple levels — Cannot correct *erroneous merges/splits* — May consider object "*linkages*"
Density-based methods	— Can find *arbitrarily shaped clusters* — Clusters are *dense regions* separated by *low-density regions* — Each point must have a *minimum number of points within its "neighborhood"* — May *filter out outliers*
Grid-based methods	— Use a multi-resolution *grid data structure* — *Fast processing time*

Partitioning Methods

1. K-Means – A Centroid-Based Technique

- Divide dataset into ***k*** **mutually exclusive** clusters

بيقسم الداتا لمجموعه من الكلستر وبيكونوا مختلفين عن بعض وانا اللي بحدد عدد cluster

- Clusters are represented by their **centroids**

كل cluster بيتمثل بالسنتر بتاعه

  - A centroid is a **cluster's center point**
- In *k*-means → centroid is **mean** of points within cluster

في k-mean السنتر بيتمثل بالنقط اللي داخل cluster

  - Each object ***x*** in cluster has a distance from centroid c_i → $dist(x, c_i)$
  - ***x*** is assigned to most similar cluster → C_i with ***min*** $dist(x, c_i)$
  - Cluster means are updated, then assignment is repeated
- To measure cluster quality → minimize sum of squared errors

$$E = \sum_{i=1}^{k} \sum_{x \in C_i} dist(x, c_i)^2$$

Factors to consider:

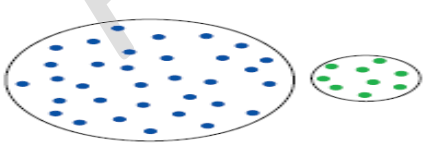
- Selection of ***k***
- Selection of ***initial centroids***
- Calculation of ***dissimilarity***
- Calculation of ***cluster means***

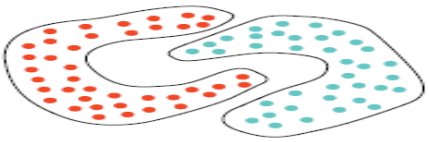
When it fails!

العوامل الموثره في الالجرويزم ده

اختياري K و انا اللي بافترض السينتر بتاع كل cluster وحساب ***dissimilarity*** و ***cluster means***

- Clusters with **very** different sizes & with **concave** shapes





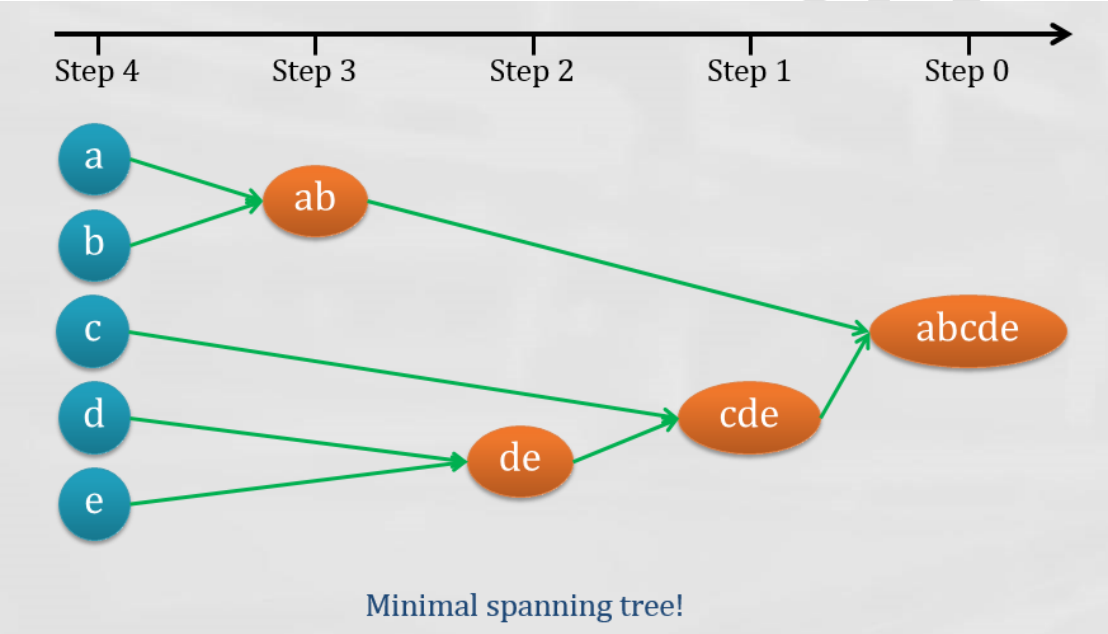
مابيطلعش cluster مختلفين في الحجم او شكلهم مقعر او مش مستوي

Hierarchical Methods

1. Agglomerative versus Divisive Clustering

- **Hierarchical clustering** → group data objects into a *hierarchy* or "*tree*" of clusters
- **Agglomerative** → bottom-up (merge) composition
  - Each object has its own cluster
  - Two clusters that are <u>closest</u> **merged** into a bigger cluster
  - Iteratively merge till *termination condition* or *single cluster* is formed

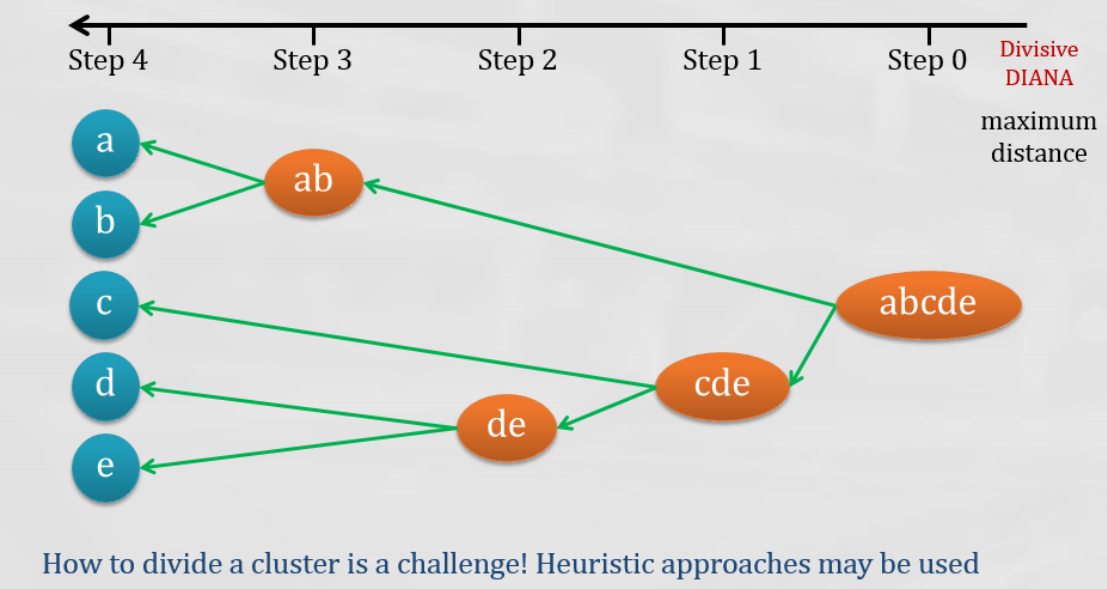
بافترض ان كل item او object وكل اتنين cluster قريبين من بعض يبقوا cluster واحد وافضل اجمع كل اتنين cluster لحد شرط معين اكون فرضته في الاول او عدد cluster معين



مثلا لو قيست المسافه بين كل نقطه والتانيه ولقيت ان a , b الاقرب لبعض وكذلك d ,e دلوقتي انا عندي ٣ cluster هاعمل نفس حكايه بيهم هاشوف المسافه بين ab و c والمسافه بين de و c لما اقيس المسافه هاخد النقطتين بتوع ab واقيس كل واحده مع c وكذلك de وهاشوف في الاخر c اقرب لاي نقطه يبقي تبع cluster اللي فيه النقطه دي وهكذا

- **Divisive** → top-down (split) composition
  - All objects in one big cluster
  - **Divide** into subclusters
  - Recursively divide subclusters into even smaller subclusters
  - Terminate when *each object has his own cluster* or *objects in clusters are similar "enough"*

هنا العكس هافترض ان كل الدانا اللي عندي عباره عن cluster واحد وابدا اقسم فيها لحد



شرط معين او لما الاقي كل object في cluster متشابهين

المشكله هنا هاقسم بناء علي ايه؟؟؟

CHAMELEON: Multiphase Hierarchical Clustering Using Dynamic Modeling

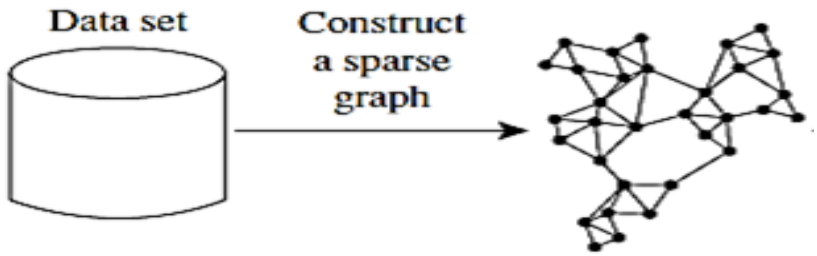
- Cluster similarity based on: مبني علي
  - **Interconnectivity** → how *well connected objects* are within a cluster

مدي ان object متوصله ببعض داخل cluster الواحد

  - **Closeness** → the *proximity* of *clusters*

مقدار التقارب بين cluster

1. Construct a **sparse graph**:
   - *Vertices* are data objects
   - there exists an *edge* between two vertices {***x***, ***y***} if ***x*** is among the *k*-most similar objects to ***y*** → *k-nearest neighbor graph*



   - Edges are *weighted* to reflect similarity

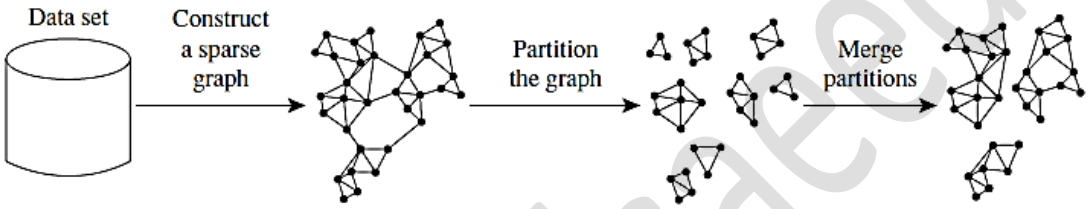
كل edge عليها weight بتاعها اللي بيمثل المساحه بين 2 node وارسم graph بناء علي قواعد معينه بيستخدم *k-nearest neighbor graph*

2. **Graph partitioning**:
   - Minimize **edge cut** $EC(C_i, C_j)$→ minimize weight of edges that would be cut to split C into C_i & C_j → measures *absolute interconnectivity*

عايز اقسم graph عشان اقلل نسبه الخطا عندي

Width عايزه اقل مايمكن فيه الجوريزمات بتعمل الكلام ده

3. **Agglomerative clustering**:
   - Measure *relative interconnectivity* $RI(C_i, C_j)$
   - Measure *relative closeness* $RC(C_i, C_j)$



بعمل graph للداتا اللي عندي وبعد كده بقسم graph ده لاشكال اصغر منه عشان اقلل نسبه الخطا وبعد كده ادمجهم ببعض بحيث اطلع cluster من كل كام graph صغير

Graph الاول ده مش انا اللي بعمله في applications جاهزه بتعمل الكلام ده.

- **Relative Interconnectivity** $RI(C_i, C_j)$ → absolute connectivity between C_i, C_j <u>normalized</u> by *internal interconnectivity* of C_i, C_j

$$RI(C_i, C_j) = \frac{\left|EC_{\{C_i,C_j\}}\right|}{\frac{1}{2}\left(\left|EC_{C_i}\right| + \left|EC_{C_j}\right|\right)}$$

where $EC_{\{C_i,C_j\}}$= sum of weights of edges that connect Ci with Cj.

  - EC_{Ci} → <u>*min sum*</u> of cut edges that partition C_i into two roughly equal parts

<u>Relative Interconnectivity->Sum</u>

- **Relative Closeness** $RC(C_i, C_j)$ → absolute closeness between C_i, C_j normalized by the internal closeness of C_i, C_j

$$RC(C_i, C_j) = \frac{\overline{S_{EC_{\{C_i,C_j\}}}}}{\frac{|C_i|}{|C_i|+|C_j|}\overline{S_{EC_{C_i}}} + \frac{|C_j|}{|C_i|+|C_j|}\overline{S_{EC_{C_j}}}}$$

- $\overline{S_{EC_{\{C_i,C_j\}}}}$ → *average weight* of edges connecting vertices in C_i to vertices in C_j

$\overline{S_{EC_{\{C_i,C_j\}}}}$ هو average weight لل edge اللي بيربط مابين

Relative Closeness->average

- $\overline{S_{EC_{C_i}}}$ →avg weight of edges belonging to **min-cut bisector** of C_i

شرح المعادلات

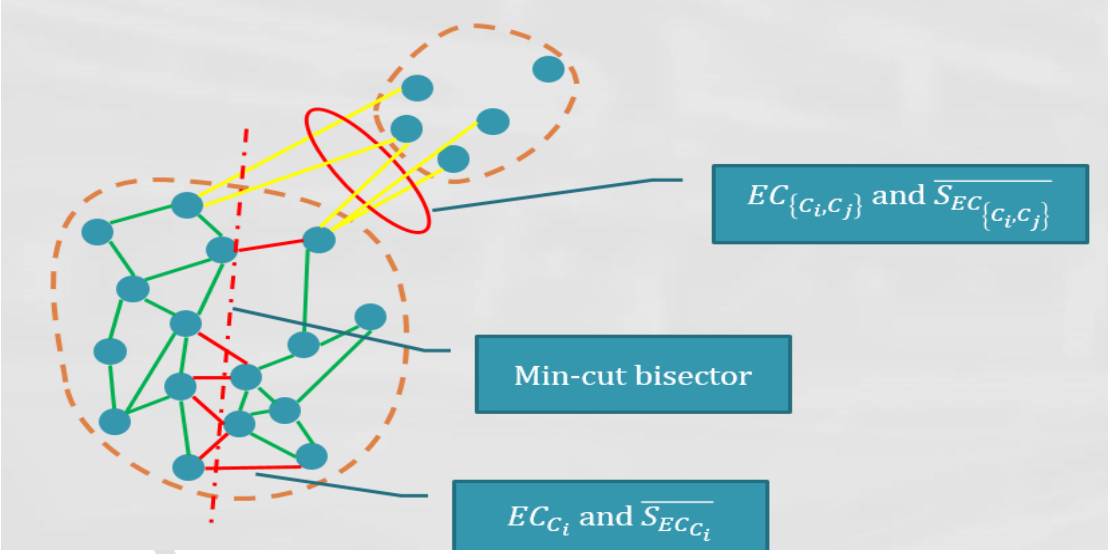
$EC_{\{C_i,C_j\}}$ *and* $\overline{S_{EC_{\{C_i,C_j\}}}}$ بيبقوا edges اللي بتربط بين 2 كلستر

$|EC_{C_i}| + |EC_{C_j}|$ or $\overline{S_{EC_{C_i}}} + \overline{S_{EC_{C_j}}}$= weighted sum of edges that partition the cluster into roughly equal parts

ودول بيبقوا edge اللي جواه cluster اللي بيقسموه الي جزئين متساوين

من الرسمه **min-cut bisector** بيقسم cluster اللي جزئين ال edge اللي بيمر بيها الخط ده هما دول $|EC_{C_i}| + |EC_{C_j}|$ or $\overline{S_{EC_{C_i}}} + \overline{S_{EC_{C_j}}}$

خلاصه المعادلات ان اللي في البسط هو مابين cluster والمقام كل cluster لوحدها



Density-based Methods

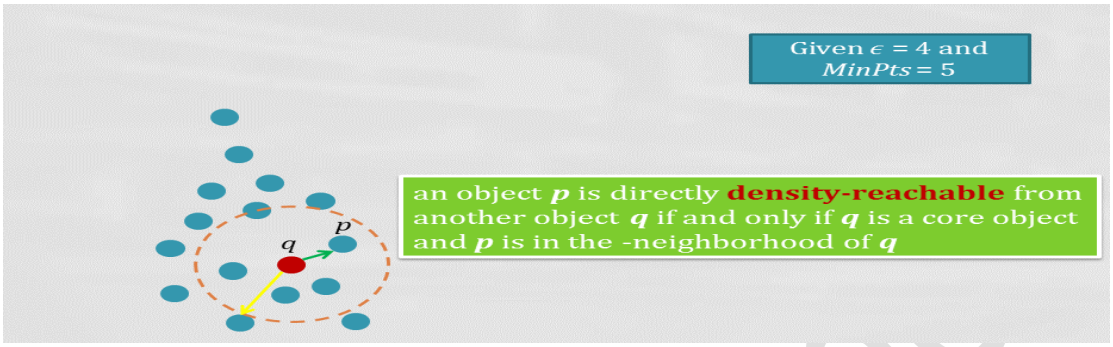
DBSCAN: Density-Based Clustering Based on Connected Regions with High Density

- Find *core objects* (with dense neighbourhoods)
- Connect core objects to form dense clusters
- User provides:
  - ***ϵ*-neighborhood** of object ***o*** →space within a radius ***ϵ*** centered at ***o*** المسافه(نصف القطر) من النقطه لاي نقطه حواليها

- **Neighborhood density** → # objects in that neighborhood
- ***MinPts*** → **density threshold** for a neighborhood اقل عدد من النقط في الكلستر الواحد

- **Core object** → object whose ϵ-neighborhood contains at least *MinPts* object

النقطه اللي اقل عدد من النقط حواليها في دائره نصف قطرها المسافه اللي محددها هو اكبر من او يساوي *MinPts*



P is **density-reachable** from ***q)*** يعني p اقدر اوصلها من q

يعني لو انا محدد المسافه 4 (نص القطر) واقل عدد في الكلستر الواحد 5 يبقي اشوف المسافه بين النقطه واللي حواليها (اللي حواليها في دائره نصف قطرها المسافه اللي

محددها وهي 4) لو في الدائره دي عدد النقط اقل من 5 يبقي اشوف نقطه تانيه وتبقي دي السينتر بتاع الكلستر واحدد من خلالها الكلستر لو فيه نقطه علي خط الدائره تتحسب انها تبعها

هاشوف كل النقط واطلع clusters اللي موجوده

q & ***m*** **density-connected** ***o*** لو المسافه مابين q و o زي المسافه بين m و o يعني اقدر اوصل لل q من o او اقدر اوصل لل m من o

مميزاته بيقدر يطلع cluster باشكال مختلفه

Evaluation of Clustering
Assessing Clustering Tendency

- Determines whether a given data set has a non-random structure
- **Hopkins Statistic** → Statistical tests for spatial randomness
  - Sample ***n*** points, $p_1, \ldots, p_n$ uniformly from ***D***
  - For each point, $\boldsymbol{p_i}$ find its nearest neighbor in ***D***

بيقارن كل dataset باقرب النقط ليها(neighbors) وبيحسب المسافه بينهم ويجيب اقربهم. لو عندي n فيها شويه نقط $\boldsymbol{p_i}$ بيشوف لكل نقطه فيهم اقرب النقط في dataset

$$distance\ x_i = \min_{v \in D}\{dist(p_i, v)\}$$

  - Sample ***n*** points, $q_1, \ldots, q_n$, uniformly from ***D***
  - For each q_i find its nearest neighbor of qi in ***D***-{q_i}

بيقارن كل dataset باقرب النقط ليها(neighbors) وبيحسب المسافه بينهم ويجيب اقربهم. لو عندي n فيها شويه نقط $\boldsymbol{p_i}$ بيشوف لكل نقطه فيهم اقرب النقط في ***D***-{q_i}

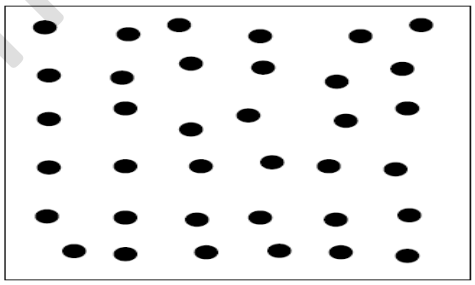
$$distance\ y_i = \min_{v \in D, v \neq q_i}\{dist(q_i, v)\}$$

  - Calculate the ***Hopkins Statistic*** ***H***:

$$H = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i}$$

  - If ***D*** is uniformly distributed, $\sum_{i=1}^{n} x_i \ and \sum_{i=1}^{n} y_i$ are roughly equal, and $\boldsymbol{H} \approx 0.5$

وبيكرر العمليه دي اكتر من مرة عشان يقدر يحسب اقرب النقط ليها



Measuring Clustering Quality – Extrinsic Methods

Extrinsic methods → compare clustering against **ground truth** (*supervision*)

النوع ده بيبقي supervised وبيتم عن طريق مقارنه الكلستر ب ground truth

- Assign a score $Q(C, C_g)$ to capture:
  - **Cluster homogeneity** → the purer the better – clusters represent separate class labels

تجانس الكلستر كل لما يكون انقي ومتجانس مع بعضه كل لما يكون افضل

- **Cluster completeness** → an object with a class label belongs to the cluster representing that class label

يكون object في class يعبر او يمثل ال class كله

يعني كل class يتمثل في مجموعه قليله من clusters

- **Rag bag** → objects that can't be merged into clusters belong to a *rag bag* – penalize a *misc. object* when put in a *pure cluster* more than in a rag bag

Object اللي ماينفعش احطها في cluster بتنتمي ل Rag bag

- **Small cluster preservation** → splitting a small category is *more harmful* than splitting a large category

اني اقسم category صغير الي اجزاء اصغر مضر اكتر من اني اقسم category كبيره

- Ex. **Bcubed** *precision* and *recall* of <u>every</u> object in dataset:
  - Precision → how many objects in the same cluster ∈ the same category as the object

كام عدد object في cluster الواحد ينتموا لنفس category

  - Recall → how many objects of the same category are assigned to the same cluster

كام عدد object في category الواحد ينتموا لنفس cluster

Intrinsic methods → measure how well the clusters are **<u>separated</u>**

النوع ده بيبقي unsupervised وغير متاح هنا بيقيم جوده الكلستر بانه **<u>ground truth</u>** بيشوف ازاي clusters منفصلين وازاي بيتم الدمج بينهم

- Ex. **The silhouette coefficient** → difference between:
  - *average distance* between object ***o*** and all other objects in the cluster to which ***o*** belongs (captures <u>cluster correctness</u>) – smaller is better (more compact)

متوسط المسافه بين object O وبين كل object اللي في نفس cluster اللي بينتمي ليها O كل لما كانت اصغر كل لما كانت افضل(العلاقه بين object في نفس الكلاس)

  - *minimum average distan*ce from ***o*** to all clusters to which ***o*** does not belong (captures <u>degree of separation</u> from other clusters) – larger is better

اقل متوسط مسافه بين object O وبين كل cluster التانيه اللي مش فيها O كل لما كانت اكبر كل لما اكنت احسن(العلاقه بين object في كلاس وبقيت الكلاسات)

- Compute average silhouette coefficient for all objects in a cluster **or** over all of the dataset
  - **+ve** → clustering is good
  - **-ve** → clustering is bad

بحسب المتوسط لل silhouette coefficient لكل dataset او لكل عنصر في cluster واشوفه لو موجب يبقي عمليه cluster كويسه لو سالب تبقي فيه مشكله في عمليه cluster